IN THE CLAIMS

Please amend the claims as follows:

1.   (Canceled)

2.   (Canceled)

3.   (Canceled)

4.   (Canceled)

5.   (Canceled)

6.   (Canceled)

7.   (Currently Amended) A software program for facilitating the use of a distributed directory running in a computer network, the program comprising being stored on a recordable medium and including instructions for:

~~receiving a first event from a first application;~~

~~converting the first event into XML data representing the first event;~~

~~transforming the XML data representing the first event to a first predetermined format by the transformation processor, the first predetermined format being responsive to a distributed directory, wherein a first portion and a second portion of the distributed directory are located in a first partition and a second partition, respectively;~~

~~transmitting the transformed XML data representing the first event to the distributed directory;~~

~~after receiving the first event from the first application,~~ receiving ~~a second~~ an event from the distributed directory into an XML generator;

converting the ~~second~~ event into XML data representing the ~~second~~ event;

transforming the XML data representing the event to a first predetermined formed by a transformation processor using a first stylesheet, the first predetermined format being responsive to a first application running in the computer network;

transmitting the transformed XML data representing the event to the first application;

transforming the XML data representing the ~~second~~ event to a second predetermined format by ~~a~~ the transformation processor <u>using a second stylesheet</u>, the second predetermined format being responsive to a second application running in the computer network; <u>and</u>

transmitting the transformed XML data representing the second event to the second ~~application;~~ <u>application.</u>

~~after receiving the first event from the first application, receiving a third event from the distributed directory into an XML generator;~~

~~converting the third event into XML data representing the third event;~~

~~transforming the XML data representing the third event to a third predetermined format by the transformation processor, the third predetermined format being responsive to a third application running in the computer network; and~~

~~transmitting the transformed XML data representing the third event to the third application;~~

~~providing a first stylesheet to the XSLT processor, the stylesheet including formatting instructions for transforming XML data to the first predetermined format; and~~

~~providing a second stylesheet to the XSLT processor, the second stylesheet including formatting instructions for transforming XML data to the second predetermined format wherein the first style sheet is different from the second stylesheet.~~

8.   (Canceled)

9.   (Previously Presented)   The software program of claim 7 further comprising instructions for:

receiving updates to the first stylesheet responsive to any changes in either the distributed directory or the first application.

10.   (Canceled)

11.   (Currently Amended) The software program of claim 7 further comprising instructions for:

detecting the ~~second~~ event through notification from an event handler of the distributed directory.

12.   (Canceled)

13.   (Canceled)

14.   (Canceled)

15.   (Currently Amended) A software program for facilitating the use of a distributed directory running in a computer network, the program comprising instructions for:

receiving ~~an~~ a first event from ~~an application prior to receiving event data from a distributed directory, wherein a first partition and a second partition of the distributed directory are located at a first physical location and a second physical location, respectively;~~ a first application in a first native application format;

converting the first event into markup language data;

transforming the first event to a predetermined format by a transformation processor using a transformation profile, the predetermined format being responsive to the distributed ~~directory;~~ directory, the transformation profile including formatting instructions for transforming the markup language data to the predetermined format;

~~providing a transformation profile to the transformation processor, the transformation profile including formatting instructions for transforming the markup language data to the predetermined format; and~~

transmitting the transformed first event to the distributed ~~directory.~~ directory;

receiving a second event from a second application in a second native application format;

converting the second event into markup language data;

transforming the second event to the predetermined format by the transformation processor using the transformation profile;

transmitting the transformed second event to the distributed directory.

16.   (Canceled)

17.   (Canceled)

18.   (Currently Amended) A distributed computer system comprising:

a first processor connected to a network for executing computer code;

a second processor connected to the network for executing computer code;

a first memory connected to the first processor;

a second memory connected to the second processor;

a distributed directory, wherein first and second portions of the distributed directory are stored in the first memory and the second memory, respectively;

~~an~~ a first application, a portion of which being stored in <u>one of</u> the <u>first memory and the</u> second memory;

<u>a second application, a portion of which being stored in one of the first memory and the second memory;</u>

a first transformation profile ~~for~~ defining a first predetermined format for use by the ~~distributed directory;~~ <u>first application;</u>

a second transformation profile ~~for~~ defining a second predetermined format for use by the <u>second</u> application;

software for detecting a directory event in the distributed directory;

~~software for detecting an application event in the application prior to detecting the directory event;~~

~~software for transforming the application event to the second predetermined format by using a generic transformation tool and the first transformation profile; and~~

~~software for providing the transformed application event to the distributed directory;~~

software for transforming the directory event to the first predetermined format by using a generic transformation tool and the first transformation profile; ~~and~~

<u>software for transforming the directory event to the second predetermined format by using the generic transformation tool and the second transformation profile;</u>

software for providing <u>to the first application</u> the ~~transformed~~ directory ~~event to the application,~~ event <u>transformed to the first predetermined format; and</u>

<u>software for providing to the second application the directory event transformed to the second predetermined format.</u>

~~whereby the distributed directory becomes aware of the application event by having the application event provided to the distributed directory in a transformed state and whereby the application becomes aware of the directory event by having the directory event provided to the application in a transformed state.~~

19.   (Previously Presented)   The system of claim 18 further comprising:

software for converting the directory event to a generic data description before transforming the directory event.

20.   (Currently Amended) The system of claim 18 further comprising:

an application shim for the <u>first</u> application to receive the transformed directory event and provide the directory event to the <u>first</u> application by using a <u>first</u> native application program interface for the <u>first</u> application.

21.   (Canceled)

22.   (Currently Amended) The system of claim 18 wherein the generic transformation tool utilizes a markup language and the software for transforming the directory event ~~and the software for transforming the application event~~ utilizes a transformation processor.

23.   (Canceled)

24.   (Canceled)

25.   (Canceled)

26.   (New) The software program of claim 7 wherein:

transmitting the transformed XML data representing the first event to the first application includes transmitting the transformed XML data representing the event to the first application through a first application shim to provide the transformed XML data representing the second event to the first application by using a first native application program interface for the first application; and

transmitting the transformed XML data representing the second event to the second application includes transmitting the transformed XML data representing the event to the second application through a second application shim to provide the transformed XML data representing the event to the second application by using a second native application program interface for the second application.

27.   (New) The software program of claim 7 wherein the first predetermined format and the second predetermined format are the same predetermined format.

28.   (New) The software program of claim 7 further:

transforming the first event to a second predetermined format by the transformation processor using a second transformation profile, the second predetermined format being responsive to the second application, the transformation profile including formatting instructions for transforming the markup language data to the second predetermined format; and

transmitting to the second application the first event transformed to the second predetermined format.

29.   (New) The system of claim 18 further comprising:

a directory transformation profile defining a directory predetermined format for use by the distributed directory;

software for detecting an application event in the first application;

software for transforming the application event to the directory predetermined format by using the generic transformation tool and the directory transformation profile; and

software for providing the transformed application event to the distributed directory.

30.   (New) The system of claim 29 further comprising:

software for detecting a second application event in the second application;

software for transforming the second application event to the directory predetermined format by using the generic transformation tool and the directory transformation profile; and

software for providing the transformed second application event to the distributed directory.

31.   (New) The system of claim 20 further comprising:

a second application shim for the second application to receive the transformed directory event and provide the directory event to the second application by using a second native application program interface for the second application.

32.   (New) A method for interfacing with a directory in a computing system, comprising:

providing a first transformation profile defining a first predetermined format for use by a first application;

providing a second transformation profile defining a second predetermined format for use by a second application;

detecting an event in the directory;

transforming the event to the first predetermined format by using a transformation tool and the first transformation profile;

transforming the event to the second predetermined format by using the transformation tool and the second transformation profile;

providing to the first application the event transformed to the first predetermined format; and

providing to the second application the event transformed to the second predetermined format.

33.   (New) The method of claim 32 further comprising the step of:

converting the event to a generic data description before transforming the event to the first predetermined format and the second predetermined format.

34. ⸜ (New) The method of claim 32 further comprising the step of:

providing an application shim for the first application to receive the event transformed to the first predetermined format and to provide the event to the first application by using a native application program interface for the first application.

35.   (New) The method of claim 34 further comprising the step of:

updating the application shim and the first transformation profile responsive to changes in the first application.

36.   (New) The method of claim 34 further comprising the step of:

providing a second application shim for the first application to receive the event transformed to the second predetermined format and to provide the event to the second application by using a second native application program interface for the second application.

37.   (New) The method of claim 36 further comprising the step of:

updating the second application shim and the second transformation profile responsive to changes in the second application.

38. (New) The method of claim 32 wherein the transformation profile includes a stylesheet.

39. (New) The method of claim 32 wherein the transformation profile is stored in the directory.

40. (New) A driver infrastructure for interfacing a directory and applications comprising:

a generator to receive a directory event from the directory and to generate a generic data for the directory event;

a first transformation profile defining a first predetermined format for use by a first application;

a second transformation profile defining a second predetermined format for use by a second application;

a transformation processor to transform the generic data for the directory event into a first application data for the first application using the first transformation profile and to transform the generic data for the directory event into a second application data for the second application using the second transformation profile; and

a transmitter to transmit the first application data to the first application and to transmit the second application data to the second application.

41. (New) A driver infrastructure according to claim 40 wherein:

the driver infrastructure further comprises a second generator to receive an application event from the first application and to generate a second generic data for the application event;

the transformation processor is operative to transform the second generic data for the application event into a directory data; and

the driver infrastructure further comprises a receiver to receive the directory data in the directory.